IN THE SPECIFICATION

Please amend page 2, lines 1-3, as follows:

--The coding system applying the inventive compression method can be also advantageously used for replacing the mechanical tape recording system of conventional video cameras, storing digital video data e.g. in Flash memory.--

Please amend page 3, lines 1-3, as follows:

--One of the most widespread and known methods for compressing video data is MPEG. It can be regarded a hybrid coding method, as it unites the compression based on spatial redundancy and the compression based on temporal redundancy.--

Please amend page 4, lines 1-4, as follows:

--The first frame has no reference, which means that it is always a so-called intra frame (I frame). Thus, with the first frame the above procedure is repeated until the entire I-type frame is processed. Frames or blocks that use the previous or subsequent frame as reference are called respectively P- and B-type frames/blocks.--

Please amend page 4, line 29, to page 5, line 7, as follows:

--Compression systems relying on temporal redundancy encode only the changed portions of consecutive frames. In practice, this is done by dividing the frames into blocks and comparing individual blocks pixel by pixel with pixels located in a search range of the previous or the subsequent frame. The procedure is illustrated in FIG. 2, showing that a reference block best matching a given block 20 of a frame 17 is being searched for in the

search range 21 located in the previous frame 16 or in the search range 19 of the subsequent frame 18. The reference block can be located anywhere, it need not coincide with the search ranges (shown in grey) designated in the reference frames 16 or 18. It may of course happen that in such cases the reference search is unsuccessful in the given reference frame(s). Evidently, the reference frames 17, 18 are not divided into blocks for the reference search, the blocks are shown in FIG. 2 only for the sake of better overview.--

Please amend page 6, lines 1-7, as follows:

--If we examine the issue in more detail, it soon turns out that the cause of excessive errors (and therefore unsuccessful searches) is the error measurement method. For instance, in case of noisy frames even in the best position the two blocks cannot be identical, the information content of the blocks is different just because of the noise. This situation also arises when the displacement of a block is not the exact multiple of the pixel size, the displacement ends somewhere between two integer pixels, i. e. the real displacement can only be exactly expressed as a fraction of the pixel size.--

Please amend page 7, line 28, to page 8, line 3, as follows:

--Next, either the resulting differences - in case of a successful search - or, if the search was unsuccessful, the current block itself are converted with DCT transformation from spatial representation to frequency representation. Then the unnecessary precision of data is reduced by the so-called quantization operation. This essentially involves discarding higher-order coefficients produced by the DCT, since these coefficients are usually small. The remaining DCT coefficients are also either small values or zeroes, which may be efficiently

coded by entropy coding, simultaneously with the position value established above. This procedure is illustrated in FIGS. 4-5.--

Please amend page 8, line 31, to page 9, line 9, as follows:

--In case a block is not an intra, but an inter coded block, then FIG. 5 shows the coding of a matched and compensated P-type block. It is sought to find a reference block 43 for the block 42 to be coded. The block 42 is located in its original position 40, between the blocks 38 of current frame to be coded. The reference block 43 may be located in the position indicated with the reference numeral 41. The search is performed by stepping the block 42 to be coded through the search range 39 of the reference frame. If the best match is found, the block 42 to be coded is subtracted from the reference block 43 (or the other way round) to generate the error between the block 42 to be coded and the matched reference block 43. In this manner, the luminance 44 and chrominance 45 components of the error are obtained. These components are subjected to DCT transformation, quantization, and run-length coding in step 46, and finally, in step 47 the run-length coded data undergo further entropy coding.--

Please amend page 10, lines 1-4, as follows:

--Fig. 6 illustrates that for a B-frame, the coding system search for a reference 49 of the block C to be coded in both the preceding frame 48 and the subsequent frame 50, finally keeping as reference either the one that produced the smallest error or the linearly interpolated average of the two.--

Please amend page 10, line 25, to page 11, line 2, as follows:

--The question may arise: why use DCT when it is only a variety of FFT? The answer

is that there is empirical evidence that DCT gives better function approximation for video encoding than FFT. This is illustrated with some concrete values shown as examples in FIG. 7. The FFT coefficients 52 and DCT coefficients 53 are produced by performing, respectively, FFT and DCT transformations on the input data 51. After quantization (that is, after discarding or truncating coefficients) the truncated FFT coefficients 54 and truncated DCT coefficients 55 are obtained. Following the inverse transformations the IFFT reconstructed data 56 and the IDCT reconstructed data 57 are obtained. Plotting the reconstructed data with the curves 58 and 59 it is seen that FFT is more sensitive to coefficient truncation. I.2.6.--

Please amend page 13, line 25, to page 14, line 14, as follows:

--The logical structure (schematic functional diagram) of the hybrid coding system according to the invention is shown in FIG. 8. Main functional units of the system are in many ways similar to the known MPEG coding system shown in FIG. 1. The input video data 60, in other words, the frames to be coded are fed into the frame scaling module 61, which, according to a number of different criteria (discussed below in detail), either reduces the size of the input frame or leaves it unchanged. The entire system is controlled by a coding control unit 62, with the exact functions thereof clarified later in the present description. Frames or blocks are coded according to intra or inter coding depending on intra/inter switch 63. Blocks are directed to the output 73 in a transformed, quantized and coded state, having passed through DCT transformation module 64, quantization module 65 and entropy encoding module 72. Reference frames needed for coding inter frames are generated by an inverse quantization module 66 and an inverse DCT module 67, from which the reconstructed

reference frames are fed into the frame store 70 through a de-block filter 69. Motion compensation, i.e. production of filtered reference frames and compensated motion information 74 (motion vector and the subtracted block) is carried out by a module designated with the reference numeral 68 (with a resolution that is adjustable between 1/2, 1/4, and 1/8 pixels). The frame store 70 stores the current reference frame, with the blocks thereof being automatically refreshed (actualized). The module 71 performs the identification of the changes, and finds that block partitioning which is best suited for tracking the changes in the frame, and the module 71 describes the best block partitioning using a Quad-tree structure (detailed below). The entropy encoding module 72 is a so called neural arithmetic compressor (see below).--

Please amend page 14, line 26, to page 15, line 2, as follows:

--The inventive compression method has an essentially hybrid nature, because it exploits both temporal and spatial redundancy. The implemented compression is based on a hierarchical block structure containing blocks of dynamically varying sizes. The reference search uses not only the frames immediately preceding and following the current frame, but further preceding and subsequent frames as well, with a maximum depth of +1 and -3 frames (that is, reference search is allowed in one following and three preceding frames). High-level motion compensation is realized by the method, with a resolution ranging from <u>½ to 1/8 pixels. The</u> entropy compressor performs optimized arithmetical coding based on multi-level prediction.--

Please amend page 16, lines 1-2, as follows:

--Now we subtract the horizontal line 77 from each horizontal line of the b<u>lock to be</u>

coded. We obtain the predicted block:--

Please amend page 17, lines 1-9, as follows:

--Thus, according to the invention, a compressibility analysis is performed on the block to be coded before carrying out the DCT transformation. Based on the compressibility analysis, the block is coded with DCT and entropy coding. In most cases, however, the compressibility analysis reveals that it is worth examining the compressibility of the block also by dividing the block to be coded into further sub-blocks. In this case, the compressibility of blocks associated to the various block partition variants is analyzed, and that partitioning is selected which promises the best potential results. Finally, after the block partitioning followed by the intra prediction, DCT transformation is carried out on the basis - of the selected, potentially most favourable block partitioning.--

Please amend page 21, lines 1-13, as follows:

--For this reason, the selection of the best predicted block is performed according to the flowchart presented in Fig. 10. Block data 79 pass through multiplexer/selector 80, which, depending upon the block size, selects the current prediction mode 81 out of those above enumerated. Selector 82b can be set by the user to direct block data 79 of the predicted block into processing module 82c either directly or through a Hadamard transform module 82a. The processing module 82c produces the absolute squared sum of the block, with comparator 83 evaluating the resulting sum. In case the value is smaller than a reference threshold value, said reference threshold is overwritten by the momentary sum, with the current prediction mode being stored together with the predicted block by processing module 84. In the

following the multiplexer/selector 80 selects the mode for the next prediction, and the whole process is repeated until all available modes in other words, prediction modes pertaining to different potential partitionings of the block are tested. At the end the best predicted block, and also the prediction mode by which it was generated, is determined.--

Please amend page 22, lines 1-3, as follows:

--After the best block partitioning and the corresponding prediction mode <u>has been determined</u> as described above, the remaining transformations (DCT . . .) are carried out and the block is coded with the entropy coding module.--

Please amend page 23, lines 1-5, as follows:

--The intrapredictive coding in one proposed aspect of the invention a<u>llows for 4 possible</u> modes using dynamically changing block sizes (I-type). Thus, during the coding of an entire I-frame it is allowed to apply blocks of different sizes (applicable block sizes are listed below). It should be noted again that the chosen mode must be indicated in the header structure of the frame. The 4 possible modes in this example are the following:--

Please amend page 24, lines 1-2, as follows:

--Because three different block sizes are applied, a method is needed for se<u>lecting the optimal</u> size.--

Please amend page 25, lines 1-2, as follows:

--TH8 and TH16 are empirical constants. As the formula shows, the "varia<u>nce" value quantifies</u> the amount of visual details in the block.--

Please amend page 26, line 1 as follows:

--Because, as it is explained below, coding of the block partitioning itself requires relatively high amount of data, it could be expedient to examine if it is worth to allow using three different block sizes. In case only two different block sizes are allowed, much less additional information has to be coded for recording block partitioning data.--

Please amend page 27, lines 1-3, as follows:

--If a sub-block is not partitioned further (has a size of 8x8 pixels), the value of the associated bit is 0. In this case the sub-block is further partitioned (into 4x4-pixel sub-blocks), the value of the associated bit is 1. For example:--

Please amend page 28, lines 1-4, as follows:

--The advantage of a block partitioning where only two block sizes are allowed is that the 5-bit QT code (the partitioning descriptor) can be replaced by a single-bit code standing for the chosen partitioning (e.g., with a basic block size of 16x16, 0 may stand for a 16.times.16-block, 1 for four 8.times.8-sized sub-blocks).--

Please amend page 31, lines 1-2, as follows:

--Thus, in case the method modifies QP, the optimal table row (optimal matrices) that corresponds to the given bandwidth, is assigned to the new quantization factor.--

Please amend page 32, lines 1-7, as follows:

--If the coefficient value of the last pair is zero, said last pair need not be coded - it is

sufficient to write the end-of-block code into the sequence. The coefficient-occurrence count pairs are written in reverse order (occurrence count:coefficient) into the data sequence to be coded. The reason is that in this manner the zero value of the occurrence count parameter can be used as an end-of-block code (without, of course, the corresponding coefficient) because, if all pairs are valid ones, no combination can occur where the occurrence count is zero, so the code can be safely applied as end-of-block code.--

Please amend page 34, lines 1, as follows:

--2. **|0|PRED|DPCM|AC|EOB or if everything is zero then |0| PRED|DPCM|EOB**--

Please amend page 35, lines 1-3, as follows:

--The physical size of UV surfaces is scaled back to half relative to Y (though this causes data loss, this loss has proved to be acceptable and does not lead to a significant decrease in quality.)--

Please amend page 36, lines 1-8, as follows:

--**V.1** As it has already been pointed out, coding of inter frames is based on temporal redundancy. This means that the current frame is compared with the previous or the subsequent frame, and only the differences between the two frames are coded. Reference search modes of the method are illustrated in Fig. 15. The following alternatives are possible: Searching only in the three preceding P-type frames (Fig. 15a); searching only in the two preceding B-type frames (Fig. 15b); searching in preceding and subsequent motion compensated references (Fig. 15c, in this case B-type frames usually cannot be used as

reference).--

Please amend page 36, line 30, to page 37, line 2, as follows:

--In case of greater displacements the probability of a successful search decreases rapidly. Another disadvantage of fast search methods is that, even if the search appears successful, it cannot be made sure that the found position is the minimum point (the position of the optimal reference block) within the search range.--

Please amend page 37, line 31, to page 38, line 3, as follows:

--If the condition for the variance of the four sub-blocks is still not true, sub-blocks are merged into 4x4 sub-blocks and the above operations are repeated with two sub-blocks with constant THV42, and, if necessary, with four sub-blocks with constant THV44 (see sub-blocks 90a, 90b of block 90), attempting to find in the latter case the four biggest-variance non-adjacent sub-blocks.--

Please amend page 38, line 28, to page 39, line 3, as follows:

--The search method is preferably fine-tuned by analyzing the obtained MSE value from other aspects as well. For instance, in case the search finds multiple positions that satisfy the conditions for a match (e.g. MSE is smaller than the maximum allowed error), and these positions are located in the same direction (on the same side) seen from the starting point, and further the error increases as the search is leaving the region of these positions, then the search can be aborted because it is highly probable that the search moves away from the optimum point.--

Please amend page 39, line 30, to page 40, line 1, as follows:

--For each sub-block with excessive error the search is repeated in a reduced search range using a block size of 8x8, starting from the position of the given sub-block. If the error still exceeds the limit, the sub-block found the best by the 8x8-search is subdivided into 4x4 sub-blocks, and the search is repeated on sub-blocks satisfying the condition SAD_4>MAXSAD4.--

Please amend page 41, lines 1-3, as follows:

--The processing of 8x8 blocks is almost identical with the processing of 16x16 blocks, with the difference that if the search is still unsuccessful at the end of processing, blocks are subdivided into sub-blocks of 4x4 pixels.--

Please amend page 42, line 31, to page 43, line 2, as follows:

--AX where X stands for the transformed block that is being processed, while A, B, and C are the surrounding blocks.--

Please amend page 45, lines 1-3, as follows:

--In case the Y block had the size of 16x16 pixels, then the UV block is sized 8x8 pixels, if the Y block is sized 8x8 then the UV block is sized 4x4 and finally, to a 4x4 Y block the corresponding UV blocks are sized 2x2.--

Please amend page 45, line 31, to page 46, line 4, as follows:

--The inventive method is a so-called arithmetic coding method. Arithmetic coding is a method known per se. The basic principle of arithmetic coding involves the modification of

the upper and lower limits of an interval (range) depending on received data of the data sequence to be coded. Arithmetic coder is truly efficient only if the distribution of incoming data is known to some extent, in other words, if it the probability estimate for the value of the next input data element is known to some extent.--

Please amend page 46, line 17, to page 47, line 1, as follows:

--The array "area[0,1]" contains two 32-bit values, namely the "area[0]" and "area[1]" variables that store the upper and lower limits of the coding interval. As it is known from the theory of arithmetic coding, the interval (area[0], area[1]) is scaled with the probe value. In known methods the probe value is usually determined as a function of the frequency of occurrence of previously arrived bits. Depending on the value of the newly received bit, either the lower or the upper limit of the coding interval is modified with the scaled value. The interval can be modified (new bits can be coded) until the difference of the upper and lower limits becomes smaller than 256. In principle, other values can also be used, but for treating the overflow, 256 appeared to be the most practical value. Accordingly, when the difference of the upper and lower limits becomes less than 256, the 8 most significant bits are written out to the output data sequence, and variables representing both the lower and upper limits are shifted to the left by 8 places.--

Please amend page 49, lines 1-14, as follows:

Instead, a fraction table of 512 elements is used (represented by the "FracTbl[sum]" variable), of which the appropriate element is selected by the sum of bit frequencies in the corresponding row of the frequency table (cf. the "sum" variable in the above algorithm). To

determine the probe value, the *f0* value in the appropriate row of the frequency table is multiplied by the value retrieved above from the fraction table, and then the product is multiplied with a constant, e.g. the product is shifted 10 bits to the right. Thus, the probe value is obtained, which will fall into the interval 0..65535, which is in turn analogous with the interval 0 . . . 1. As the fraction table contains 512 elements (actually the appropriately scaled values of 1/sum, where the "sum" variable is used as an index to the table at the same time), it should be made sure that the "sum" value does not exceed this value.--

Please amend page 50, lines 1-6, as follows:

--The method presented above is a fast integer-arithmetic variety of known methods. With parameters tuned appropriately, compression efficiency is 10% higher than that of the VLC method used in MPEG systems. So far, only methods using far more complex probe algorithms have performed significantly better than that. To improve efficiency, the frequency table should also be significantly sized up. Both the chosen probe algorithm and frequency table size affect the execution time of the method.--

Please amend page 50, line 31, to page 51, line 8, as follows:

--Known coding methods, primarily the **PPM** (prediction by partial match) method examines symbol combinations of varying length. When a received symbol is coded, first the longest allowed combination is tested. The newly arrived symbol is added to the stored symbol group, and a search is performed with the current symbol group length to establish if the current group has already occurred. For instance, if the group length is 4, then the three most recent symbols will be tested together with the newly arrived one. If the symbol combination has already occurred, it is coded using the momentary or constant probability

value assigned to that given symbol combination. If, on the other hand, the combination has not yet occurred, an escape symbol is coded to indicate (for the decoder) that the combination is new, and the search is carried on with a shorter combination length.--

Please amend page 52, lines 1-4, as follows:

--Mathew V. Mahoney (Florida Institute of Technology) took over the application of neural network technology to binary arithmetic coding (where only 0 and 1 are the symbols to be coded), using the on-line training method known from neural network theory and applying adaptive learning rate instead of a constant one.--

Please amend page 53, line 31, to page 54, line3, as follows:

--Because the factor k is assigned a value by the output of the neural network and the actual lengths H1-H4 of the subdomains 152-154 are determined by k (indirectly, through union sizes, because union sizes affect the length of subdomains), the partitioning of table 155 into subdomains 151-154 is changing dynamically according to the output of the neural network after each received bit.--

Please amend page 54, line 28, to page 55, line 4, as follows:

--The probe value is 0 (corresponding to the probability 0.5) when the first bit arrives. Then the network calculates the error (error=bit-probe), and "teaches" the error to neurons assigned to the previous bit value. (In the first step these values are irrelevant. Because there are no previous data, all addresses are 0 so the 0-th neuron will be assigned to the input). Next, the system generates new addresses from the current value of the register (buffer).

Weight functions (zero in the first step) of neurons selected by the addresses are then summed up and the exponential of the sum is calculated (the result in the first step is zero as well), which becomes the new probe value.--

Please amend page 56, lines 1-5, as follows:

--**VI.5** Next, the partitions of the register 150 are tested iteratively to select the best partition. The neural network updates frequency data f(0), f(1) of the most recently addressed table rows based on the value of the next received bit, and "teaches" to the neuron weight functions stored in these rows the last value of k and the probe factors derived from frequencies f(0), f(1) with regard to the difference (error) between the predicted and received bit value.--

Please amend page 57, lines 1-4, as follows:

--**VII.1** Bandwidth (transfer rate) control is one of the most important issues in video encoding. The information content of frames in a video frame sequence varies to a great extent, so in case the aim is to maintain a substantially even image quality, and the compression ratio has to be adjusted over a large scale to follow these changes.--

Please amend page 58, lines 1-9, as follows:

--It is preferable that the signal-to-noise ratio between the original and reconstructed frames be also taken into account as a control parameter, that is, the transfer rate should be increased (within the specific limits) in case the SNR deteriorates and the transfer rate may be lowered if the SNR improves. This is the so-called variable bit rate (**VBR**) method. A major drawback of this solution is that the total expected data length cannot be predicted exactly.

Minimum and maximum values cannot be set too high, because then the control range would also be too wide and the total coded data length would vary over a too large scale. It also often happens that the desired quality cannot be maintained with the maximum transfer rate set by the system, making it necessary to further increase the transfer rate.--

Please amend page 60, lines 1-7, as follows:

--As seen in Fig. 18c, in addition to the output 188 determining the quantization factor, the neural network used in VBR mode may also comprise a further output 189 representing the scaling factor S (the role of the latter is described later). Similarly to the above described case, the network processes the input data of expected/coded quality and expected/coded length in a time sequence during training, and estimates the sought mapping in accordance with the specified minimum and maximum values. Training data are chosen to reflect the specified control characteristics and control slope.--

Please amend page 60, line 28, to page 61, line 3, as follows:

--In their practical implementation, there is no significant difference between the VBR and CBR networks, except for the input data, which means that the network performing VBR mode can perform the functions of the CBR mode as well. For CBR-mode operation, that is achieved by simply providing a constant value at the quality inputs (at the maximum possible value, which inputs are kept constant during training as well). In CBR mode the minimum and maximum bandwidth limit inputs are set equal and are kept constant, set to values corresponding to the desired constant bandwidth.--

Please amend page 62, lines 1-2, as follows:

--where Vk_n is the normalized expected/coded length ratio and C_{vk} is the generated address, and where M_n is the normalized quality and C_m is the generated address.--

Please amend page 62, line 25, to page 63, line5, as follows:

--This system operates flawlessly in multiple-step mode as well. The essence of this mode of operation is that in a first step, the system encodes the entire footage with a constant quantization factor (e.g. with Q set to 3) without control. In the subsequent second step coding is performed with the control system activated. This solution provides improved-precision coding because the first step specifies the degree to which each frame can be compressed, so Q need not be determined, but may be directly adapted from step 1. Otherwise, the inventive neural network can be applied without any modifications. In multiple-step mode training can be performed using fast-training procedures. Also, interpolation is highly effective in this mode: we have observed that the quality achieved in 4-6 steps by discrete control systems can be reached by the neural control system in as little as two steps.--

Please amend page 63, line 25, to page 64, line 2, as follows:

--VIII.2. In order to eliminate. the problems described above, in accordance with the present invention, the concept of dynamic scaling has been introduced. This essentially means scaling down (re-scaling) if the control system is unable to maintain the desired image quality due to fixed external boundary conditions. The frames are scaled down (re-sized) to a size that provides satisfactory results. The system compresses this reduced-size frame and, at decompression, restores it to its original size. Understandably, image quality deteriorates in

this case as well, however, this will primarily appear as reduced sharpness. Blocking artefacts and other typical errors caused by the compression do not arise, at least if the compression ratio is not set extremely high.--

Please amend page 64, line 21, to page 65, line 2, as follows:

--VIII.3. Accordingly, in the dynamic scaling method according to the invention, we need scaled images. A number of interpolation-based frame scaling methods were tested. The Lancos method yielded the best results (the Lancos method is a resampling procedure known per se that interpolates the missing pixel by a filter, based on spatial frequency components of the image). If compression with and without scaling are compared, it turns out that without scaling, in critical sequences the quality loss can be easily perceived if the stream is compressed for a transfer rate of 0.5 Mbit/s. Many areas in the image become completely "flat", blocking artefacts and stripes appear, with image sharpness being drastically reduced in some areas as if an eraser was applied to the image. On the other hand, in case the compression is performed with the frame scaling according to the invention, none of these errors occur. The only perceptible error is the reduction of sharpness. However, having analyzed the sequences, it was found that scaling is typically needed at those points where fast motions occur in the video footage. Because fast-moving scenes are usually slightly blurred in the original already, the information loss caused by re-scaling is barely perceptible.--

Please amend page 66, lines 1-2, as follows:

--X. Some general remarks concerning the neural control system applied for the

present invention--

Please amend page 67, lines 1-2, as follows:

--XI.1. A summary of the operation of the hybrid video coding system implementing the inventive methods--

Please amend page 68, lines 1-7, as follows:

--A drawback of the inventive coding system is that, due to arithmetic coding, it is sensitive to errors caused by data loss in the transmission channel. However, contemporary digital transmission networks (such as the Internet) are capable of high-security and substantially loss-free data transfer, even for very high amounts of data so this drawback is not significant. For operation of the coding system with good efficiency, the frequency table should be updated continuously. If a transmission error occurs somewhere during the decoding process, then from that point on all data until the end of the affected frame will be damaged.--

Please amend page 68, line 28, to page 69, line 3, as follows:

--In case of an inter frame, first the Quad-tree structure describing block partitioning is decoded at step 123, because this tree structure contains the data needed for the reconstruction of the block. These data are used for decoding DCT coefficients, motion vectors, and prediction codes associated to individual sub-blocks, and also for the decoding of codes identifying the reference frames that were used for coding. Inverse transformations are also carried out (steps 127, 128, 129), and then those blocks of the reference frame stored in reference memory 125, which blocks were selected using the motion vectors in step 124, are

added to the inverse transformed blocks in step 130.--

Please amend page 69, line 30, to page 70, line 5, as follows:

--The inventive video coding system is capable of digitizing, efficiently coding and storing video signals. At the same time, it is also capable of transcoding already encoded digital video data for increased storage efficiency. For instance, such transcoding can be applied for reducing the bandwidth of ***MPEG*** transport packets of a DVB broadcast from approx 20 Mbit/s to approx. 600 Kbit/s, e.g. for recording satellite or television broadcasts. In a similar manner, the inventive high-efficiency coding method can also be used for storing, video sequences recorded with digital video cameras, even without the application of mechanical devices.--